ISSN 2007-9737

Clasificación temática automática exhaustiva del corpus Reuters 21578 con aprendizaje automático supervisado

Juan Manuel Arengas Acosta, Rafael Guzmán Cabrera*,
Misael López Ramírez, Anderson Smith Florez Fuentes

Universidad de Guanajuato,
Departamento de estudios multidisciplinarios,
México

{jm.arengasacosta, guzmanc, lopez.misael, as.florezfuentes}@ugto.mx

Resumen. La clasificación automática de textos se ha consolidado como una disciplina de investigación que fusiona técnicas avanzadas de procesamiento de lenguaje natural (PLN) con algoritmos de aprendizaje automático, permitiendo categorizar eficientemente grandes volúmenes de documentos textuales. Se propone un enfoque innovador que integra técnicas actuales de preprocesamiento con algoritmos clásicos de aprendizaje supervisado para mejorar la precisión en la clasificación del corpus Reuters-21578. Se plantea una revisión literatura, la implementación de técnicas de preprocesamiento (tokenización, lematización, eliminación de stopwords, conversión a minúsculas y eliminación de caracteres especiales), al igual que la exploración de algoritmos de aprendizaje supervisado (Regresión Logística, Máquinas de Soporte Vectorial, Naïve Bayes, Random Forest y k-vecinos más cercanos). Se realizaron experimentos con diversas configuraciones, combinando técnicas de preprocesamiento, métodos de selección de características como TF-IDF, y los algoritmos ya mencionados. Es así como los hallazgos en los escenarios experimentados revelan la integración de estas técnicas y algoritmos mejora significativamente la precisión de la clasificación de textos, dando como resultado una configuración apta para el corpus Reuters-21578 que presenta una precisión de hasta el 98.6%. Se propone una metodología empírica rigurosa y eficaz, que puede ser aplicable a diversos corpus de documentos en formato de texto.

Palabras clave. Algoritmos de clasificación, procesamiento del lenguaje natural (PLN), corpus Reuters-21578, clasificación temática exhaustiva.

Automatic Thematic Exhaustive Classification of the Reuters 21578 Corpus Using Supervised Machine Learning

Abstract. Automatic text classification has established itself as a research discipline that merges advanced natural language processing (NLP) techniques with machine learning algorithms, allowing to efficiently categorize large volumes of textual documents. An innovative approach is proposed that integrates current preprocessing techniques with classical supervised learning algorithms to improve the classification accuracy of the Reuters-21578 corpus. A literature review, the implementation of preprocessing techniques (tokenization, lemmatization, stopword elimination, lowercase conversion and special character elimination), as well as the exploration of supervised learning algorithms (Logistic Regression, Support Vector Machines, Naïve Bayes, Random Forest and k-nearest neighbors) are proposed. Experiments were conducted with various configurations, combining preprocessing techniques, feature selection methods such as TF-IDF, and the aforementioned algorithms. Thus, the findings in the experimented scenarios reveal that the integration of these techniques and algorithms significantly improves accuracy of text classification, resulting in a configuration suitable for the Reuters-21578 corpus that presents an accuracy of up to 98.6%. A rigorous and efficient empirical methodology is proposed, which can be applicable to various document corpora in text format.

Keywords. Classification algorithms, natural language processing (NLP), Reuters-21578 corpus, exhaustive thematic classification.

1. Introducción

En la era digital actual, donde la información se multiplica a un ritmo sin precedentes, la clasificación automática de textos se ha consolidado como una disciplina de investigación que fusiona técnicas avanzadas de procesamiento de lenguaje natural (PLN) con algoritmos de aprendizaje automático. Esta fusión permite categorizar grandes volúmenes de documentos en formato de texto de manera eficiente, respondiendo a la creciente necesidad de buscar, evaluar, seleccionar, gestionar y analizar datos en diversos campos [1].

El presente estudio propone un enfoque innovador que integra técnicas actuales de preprocesamiento con algoritmos clásicos de aprendizaje supervisado, para mejorar la precisión en la clasificación del corpus Reuters-21578. Este corpus, ampliamente utilizado en la investigación de PLN, presenta desafíos particulares debido a la diversidad de temas y estilos de escritura que contiene [2].

Las técnicas de preprocesamiento implementadas incluyen tokenización, lematización, eliminación de stopwords, conversión a minúsculas y eliminación de caracteres especiales. Estas técnicas, como señalan [3] son fundamentales para mejorar la calidad de los datos de entrada en tareas de PLN.

Paralelamente, se exploraron e implementaron varios algoritmos de aprendizaje supervisado, incluyendo Regresión Logística, Máquinas de Soporte Vectorial, Naïve Bayes, Random Forest y k-vecinos más cercanos. La selección de estos algoritmos se basó en su eficacia demostrada en tareas de clasificación de textos, como destacan [4] en su revisión exhaustiva de algoritmos de clasificación de textos.

Posteriormente, utilizando el programa desarrollado, se realizaron experimentos con diversas configuraciones, combinando las técnicas de preprocesamiento mencionadas, métodos de selección de características como TF-IDF, y los algoritmos supervisados. Este enfoque sistemático está en línea con las recomendaciones de [5], quienes subrayan la importancia de una selección adecuada de características y algoritmos en tareas de clasificación de textos.

Un aspecto fundamental de la metodología empleada es el uso de la validación cruzada, que contribuye significativamente al desarrollo de modelos más robustos y con mayor capacidad de generalización. Esta técnica, como señalan [6], es esencial para evaluar de manera confiable el rendimiento de los modelos de clasificación.

Los hallazgos de esta investigación revelan que la integración y combinación de diversas técnicas de preprocesamiento, métodos de selección de características y algoritmos clásicos de aprendizaje supervisado no solo potencia significativamente las capacidades actuales de clasificación, sino que también sienta bases para futuras trabajos de investigación en los campos del PLN y el aprendizaje automático. En el contexto específico de este estudio, se identificó una configuración de combinación apropiada de estas técnicas y algoritmos, con la que se logra obtener la mejor precisión en la clasificación del corpus Reuter 21578.

La clasificación automática de textos se considera un área fundamental que proporciona soluciones para la gestión sistematizada de la información. Por tanto, este proyecto aborda una visión comprensiva de las estrategias utilizadas en la implementación para la clasificación de documentos de texto. Donde se exploran métodos que abarcan tanto la preparación del texto, la transformación de la información como la implementación de algoritmos de aprendizaje para realizar estas tareas.

La clasificación automática de documentos es un proceso mediante el cual se asigna automáticamente una categoría o etiqueta a un documento en función de su contenido, estructura o metadatos. Este proceso se realiza utilizando técnicas de procesamiento de lenguaje natural (PLN), aprendizaje automático (machine learning) y minería de datos, con el objetivo de organizar y gestionar grandes volúmenes de información de manera eficiente.

La clasificación automática de documentos se puede realizar de diversas maneras: puede ser por ejemplo por entidades (nombres de personas, organizaciones, ubicaciones, fechas, etc.). también se puede clasificar por prioridad (urgente, importante o rutinario) entre otras opciones. En este trabajo realizamos clasificación temática de

documentos, es decir, se enfoca en categorizar documentos según su tema o contenido.

En el marco de esta investigación, se ha realizado una revisión bibliográfica que analiza 16 estudios significativos publicados entre 2014 y 2023. Es importante destacar que estos estudios se centran en algoritmos de aprendizaje automático supervisado y técnicas de Procesamiento del Lenguaje Natural (PLN) aplicadas al corpus Reuters-21578; un conjunto de datos estandarizado de referencia en el PLN que ha sido objeto de numerosos estudios que buscan mejorar la precisión y eficiencia de la clasificación de textos y que también hace parte del objeto de estudio de este trabajo. Derivado de esta revisión, en la sección que sigue, se teje una narrativa que resalta la pertinencia de los estudios individuales y fusiona sus hallazgos, delineando el estado del arte del PLN y sentando las bases para futuros proyectos de investigación en la clasificación automática de textos.

En conclusión, esta investigación constituye una contribución notable en el campo de la clasificación automática de textos. El estudio propone una metodología empírica caracterizada por su rigurosidad y alta eficacia, cuya aplicabilidad se extiende a diversos corpus de documentos en formato de texto. La aproximación innovadora desarrollada en este trabajo no solo amplía el horizonte de posibles aplicaciones, sino que también establece una base sólida para futuras investigaciones en el área. Su versatilidad se manifiesta en la posibilidad de adaptación a variados conjuntos de datos de texto, permitiendo la identificación de la mejor configuración en la combinación de técnicas y algoritmos.

Es importante destacar que esta investigación ha dado lugar a varios estudios subsecuentes, publicados en congresos y revistas indexadas. Estos trabajos profundizan en diversos aspectos de la clasificación automática de textos, incluyendo el análisis de la influencia de las técnicas de preprocesamiento en la precisión de la clasificación, el impacto del preprocesamiento en modelos específicos como Naïve Bayes y Random Forest, y la evaluación comparativa de varios métodos de aprendizaje automático. Estos estudios derivados, que se discutirán en detalle en el capítulo de conclusiones, no solo validan la relevancia y el impacto de esta investigación, sino que también abren nuevas líneas de indagación en el campo del PLN y la clasificación automática de textos.

Como resultado de esta búsqueda, la selección inicial de documentos se compuso de 25 estudios, que luego fue depurada a 16. Debido a que, cada uno de estos estudios fue sometido a un análisis crítico, evaluando su contribución al conocimiento existente y su relevancia para el corpus Reuters-21578 y las técnicas de preprocesamiento de texto. Este proceso minucioso revela una brecha significativa en la literatura disponible, donde la fusión de técnicas específicas de PLN y algoritmos de aprendizaje automático supervisado representa un campo fértil para la investigación.

En este sentido, este estudio pretende llenar dicho vacío al ofrecer una nueva perspectiva sobre cómo la integración de estas tecnologías puede avanzar en la mejora de la clasificación de textos, especialmente para el conjunto de datos estudiado. Una vez establecida esta base y detallado el marco teórico, la investigación puede avanzar en la comprensión y mejora de las técnicas de PLN para el aprendizaje automático, posicionándose como una contribución oportuna en la ciencia de datos.

En [7] se examina la eficacia de las SVM en la clasificación de textos, por medio de la representación vectorial de textos, como la Bolsa de Palabras (BoW) y TF-IDF, y emplea métricas de rendimiento estándar para evaluar los clasificadores., Este estudio complementa las investigaciones realizadas por [6-8] en cuanto al uso de algoritmos de aprendizaje automático en el PLN. Aunque difiere en los métodos de representación y las métricas de evaluación utilizadas, una comparación directa con estos estudios podría enriquecer la comprensión de las fortalezas relativas de las SVM en la clasificación de textos.

En [6] se presenta una revisión exhaustiva del PLN aplicado al análisis de grandes conjuntos de datos, destacando la implementación de algoritmos de aprendizaje automático como el Naïve Bayes para la categorización de textos. Se examinan técnicas como la Bolsa de Palabras (BoW) y TF-IDF. A pesar de su análisis profundo, el estudio podría expandirse para incluir un espectro más amplio de algoritmos de aprendizaje

profundo que están ganando prominencia en el análisis de “big data”.

De la misma manera, [9] realiza una revisión sistemática que se centra en algoritmos de aprendizaje automático supervisado como árboles de decisión, Naïve Bayes y SVM. El estudio se basa en una búsqueda sistemática utilizando la metodología PRISMA, destacando la prevalencia y eficacia de estos algoritmos en la clasificación de datos.

Por otro lado [10] realiza un analisis comparativo de los algoritmos de Regresión logística, Random Forest y KNN, aplicando técnicas de preprocesamiento y vectorización como TF-IDF a un conjunto de datos de noticias. Se demuestra que la regresión logística supera a los otros algoritmos en términos de precisión. La preferencia por la regresión logística encontrada en este estudio se hace eco de los hallazgos de [5], aunque con un enfoque más estrecho en un solo tipo de conjunto de datos. Integrar los hallazgos de [10] con estudios que utilizan una gama más amplia de datos podría proporcionar una visión más matizada de la aplicabilidad de la regresión logística en la clasificación de textos.

En [11] se explora la selección de características y el rendimiento de la clasificación utilizando modelos como Random Forest, SVM y KNN. Los conjuntos de datos variados utilizados en el estudio proporcionan una base para evaluar la eficacia de estas técnicas en diferentes contextos. Al igual que [10], este estudio ofrece una perspectiva más amplia en cuanto a la variedad de conjuntos de datos. Esta comparativa resalta la flexibilidad del Random Forest no solo como clasificador sino también como herramienta para la selección de características.

Desde otra perspectiva el aplicar técnicas de clasificación de textos en medios sociales, destacando la relevancia de algoritmos como Random Forest y SVM. [12] utiliza un conjunto de datos extraídos de X (antes Twitter) para evaluar precisión y recall en la detección de sentimientos y opiniones. Mientras que [12] abordan de modo eficiente la clasificación de textos cortos y ruidosos como los típicos de los medios sociales, el estudio podría expandirse para considerar la influencia de las técnicas de PLN específicas para este tipo de texto, como el manejo de emojis y abreviaturas. Este trabajo se alinea con los estudios anteriores en términos de métodos de evaluación de clasificadores, pero añade el elemento único de los textos de medios sociales. Sería interesante comparar los métodos aplicados en este contexto con aquellos utilizados en textos más largos y formales.

Así mismo, [4] ofrece una revisión completa de los métodos de clasificación de texto, discutiendo algoritmos y técnicas desde preprocesamiento hasta evaluación de clasificadores. Se realiza una revisión en un amplio rango de técnicas y algoritmos brindando una visión panorámica del campo de la clasificación de texto. Sin embargo, sería valioso explorar la aplicación de estas técnicas en contextos de aprendizaje en línea y flujos de datos continuos. [4] complementa los estudios de [11-13], proporcionando una base teórica más sólida sobre la que es posible aplicar las técnicas de clasificación.

Por otro lado, [11] compara varias técnicas de representación de texto, como one-hot encoding, bolsa de palabras y embeddings de palabras, y su impacto en la clasificación de textos. la comparativa de [11] es invaluable para entender cómo diferentes representaciones influyen en el rendimiento del clasificador. Aunque el estudio es exhaustivo, incluye los más recientes avances en embeddings contextuales como BERT o GPT podría beneficiarle.

[14] Investiga el impacto de la lematización como técnica de preprocesamiento en la clasificación de textos, comparando el rendimiento de varios clasificadores antes y después de la lematización utilizando conjuntos de datos estándar. El enfoque presentado en el estudio es la lematización en el preprocesamiento de texto necesario. Sin embargo, el alcance de la investigación podría incluir otras técnicas de normalización de texto y su efecto comparativo para ser más amplio.

[15] explora el impacto del aprendizaje profundo en la clasificación de textos, poniendo especial énfasis en redes neuronales convolucionales y recurrentes, además de presentar una comparación de estos métodos con clasificadores más tradicionales en varios conjuntos de datos.

[16] centra el análisis de sentimiento en comentarios de productos en línea empleando técnicas de clasificación como Naïve Bayes y SVM

para determinar la polaridad de los comentarios, con un enfoque en la optimización de características para mejorar la precisión. El enfoque práctico presentado evidencia una aplicación directa de técnicas de NLP, pero se beneficiaría de proporcionar mayor atención a los sesgos potenciales en los datos y en cómo estos afectan el rendimiento del clasificador.

[17] examina el uso de redes neuronales en la clasificación automática de noticias, evaluando distintas arquitecturas de redes neuronales y su efectividad en comparación con los métodos tradicionales de clasificación. Llevan a cabo un análisis riguroso de las capacidades de las redes neuronales, aunque la inclusión de un análisis de error detallado hubiera sido útil para entender las limitaciones de estos modelos en contextos específicos.

[18] Abordan los retos asociados con la clasificación de textos multilingües, poniendo a prueba una gama de algoritmos de clasificación, incluyendo técnicas de aprendizaje profundo, para evaluar su eficacia en conjuntos de datos en varios idiomas.

[19] optimiza los hiperparámetros para mejorar el rendimiento de clasificadores de texto, comparando diferentes métodos de optimización en algoritmos como SVM y Random Forest. Ofrecen una visión profunda sobre la importancia de la optimización de hiperparámetros. Sin embargo, sería valioso extender el estudio a clasificadores de aprendizaje profundo, donde la optimización puede ser más compleja y crítica.

Por otro lado, [11] Aborda la preocupación en torno a la desinformación en línea, evaluando la efectividad de varios algoritmos de clasificación para identificar noticias falsas, incluyendo Naïve Bayes, SVM y redes neuronales. El enfoque en la desinformación complementa los estudios previos centrados en análisis de sentimientos y clasificación de noticias, como los realizados por [16] extendiendo la aplicabilidad de NLP a retos sociales actuales.

[20] Detecta spam en correos electrónicos y plataformas en línea, aplicando algoritmos como Random Forest y Naïve Bayes para filtrar mensajes no deseados. Relación con otros estudios: Este estudio se sitúa en el ámbito práctico del PLN, similar al análisis de sentimiento de [16], pero con un enfoque diferente. La convergencia de métodos de detección de spam y noticias falsas sugiere una posible sinergia en futuras investigaciones.

Ahora bien, considerando el rigor metodológico y el análisis empírico que se lleva a cabo en esta investigación, la meta de mejorar la precisión de clasificación en al menos un 1% es ambiciosa pero alcanzable. La aplicación cuidadosa y crítica de las técnicas y algoritmos identificados en la revisión proporciona una sólida base teórica para el diseño e implementación del programa propuesto.

Además de los estudios ya mencionados, es importante destacar la evolución de las técnicas de procesamiento de lenguaje natural y aprendizaje automático en los últimos años. Por ejemplo, estudios recientes han explorado el uso de modelos basados en transformadores, como BERT y GPT, que han demostrado una notable mejora en la precisión y comprensión contextual de los textos. Aunque este estudio se centra en algoritmos clásicos de aprendizaje supervisado, futuros trabajos podrían beneficiarse de la integración de estos modelos avanzados para comparar su rendimiento con los métodos tradicionales.

Finalmente, traer a la memoria que este estudio busca contribuir al campo de investigación, desarrollando un programa en Python, que integre técnicas de preprocesamiento y selección de características con algoritmos de aprendizaje automático supervisado que permita seleccionar la configuración con la mejor combinación, donde se obtiene una mejora en la precisión de la clasificación de textos del corpus Reuters-21578.

2. Clasificación

2.1 Clasificación de textos

La clasificación de textos consiste en asignar etiquetas o categorías temáticas a documentos textuales, de acuerdo con un grupo preestablecido de clases. Es decir, el proceso implica inferir a cuál o cuáles de las categorías disponibles pertenece el contenido de un texto, basándose en sus características. Esta tarea permite organizar grandes volúmenes de información en formato de texto según su temática, facilitando su indexación y recuperación. Los modelos de clasificación se

entrenan con datos previamente etiquetados para aprender a asignar documentos nuevos a las clases correspondientes de forma automática [21].

La clasificación de textos se puede definir así, dado un conjunto de documentos D y un conjunto de clases C, la función objetivo Φ asigna cada documento a una o varias clases de forma binaria. La clasificación de textos es un desafío de aprendizaje supervisado en el cual, dado un conjunto de documentos y categorías, se intenta desarrollar una función que asigne cada documento a una o más clases de la manera más precisa posible.

Se puede definir formalmente de la siguiente manera:

Sea:

- $D = \{d_1, d_2, ..., d_n\}$ el conjunto de documentos a clasificar,
- $C = \{c_1, c_2, ..., c_k\}$ el conjunto de posibles clases o categorías,
- Entonces, la clasificación de textos consiste en definir una función:
- $\Phi: D \times C \rightarrow \{0,1\}$,

  Donde:
- $\Phi(d_i, c_j) = 1$ significa que el documento di pertenece a la categoría c_j,
- $\Phi(d_i, c_j) = 0$ significa que el documento d_i NO pertenece a la categoría c_j.

Es decir, la función Φ asigna cada documento d_i a una o más categorías c_j de forma binaria (pertenece o no pertenece).

En la práctica, esta función Φ no se conoce a priori y debe ser aproximada por un clasificador automático entrenado con un conjunto de documentos previamente clasificados. El objetivo es que el clasificador asigne las categorías a nuevos documentos de la forma más precisa posible.

2.2 Clasificación automática de textos

La clasificación automática de textos busca emular la pericia de un especialista humano al categorizar documentos según su temática [1, 16, 21, 22, 23, 24]. Es decir, apunta a que las etiquetas o clases asignadas computacionalmente se acerquen en gran medida a las que designaría una persona entendida tras analizar manualmente el contenido.

Para esto, los algoritmos de clasificación primero se entrenan con datos previamente rotulados por expertos, para poder construir modelos capaces de mapear características de texto a categorías específicas imitando ese discernimiento especializado.

Luego, el sistema aplica las complejas asociaciones aprendidas para replicar de forma automática la labor de clasificación sagaz que llevaría a cabo el ojo experto humano. La clasificación automática de textos consiste en estimar probabilidades de pertenencia a categorías mediante modelos entrenados con datos previamente etiquetados.

Se puede definir la clasificación automática de textos de la siguiente manera:

Sea:

- Un conjunto de documentos $D = \{d_1, d_2, ..., d_n\}$
- Un conjunto de categorías o clases $C = \{c_1, c_2, ..., c_m\}$

Se define una función de clasificación automática φ:

$$\varphi: D \times C \rightarrow [0,1],$$

donde:

- $\varphi(d_i, c_j)$ es la probabilidad estimada de que el documento d_i pertenezca a la categoría c_j,
- Los valores de φ están en el intervalo [0,1], representando el grado de pertenencia del documento a la categoría.

El objetivo de un clasificador automático es aproximar lo mejor posible la función de clasificación ideal Φ:

$$\Phi: D \times C \rightarrow \{0,1\},$$

donde:

- $\Phi(d_i, c_j) = 1$ si el documento d_i pertenece a c_j,
- $\Phi(d_i, c_j) = 0$ en caso contrario.

Es decir, la clasificación automática trata de estimar una función de probabilidad de pertenencia a categorías, acercándose lo máximo posible a la función real que solo puede tomar valores 0 o 1.

La función Φ se evalúa con conocimiento de expertos, mientras que φ se construye con técnicas automáticas de aprendizaje.

Para entrenar un clasificador automático se utiliza un conjunto de documentos previamente clasificados manualmente para ajustar los parámetros del modelo y aproximar φ a Φ. Luego se utiliza en nuevos documentos no clasificados.

Una vez construido, el clasificador puede utilizarse para asignar clases a nuevos documentos o para reevaluar documentos dado una nueva clase.

2.3 Aprendizaje automático supervisado

El aprendizaje automático, un campo definido por el uso de algoritmos y modelos estadísticos para enseñar a los sistemas informáticos a aprender y mejorar a partir de datos sin programación explícita, se centra en incrementar la precisión en tareas específicas a través del aprendizaje de estos datos [25]. Dentro de este ámbito, los algoritmos se dividen en dos categorías principales: los no supervisados, que procesan datos no etiquetados, y los supervisados, que trabajan con datos etiquetados.

Específicamente en el aprendizaje supervisado, los sistemas aprenden de un conjunto de datos etiquetados en la fase de entrenamiento para hacer predicciones sobre nuevos conjuntos de datos no etiquetados. Un desafío notable en el aprendizaje automático es la gestión de datos inciertos o ambiguos, aspecto en el que la teoría de la probabilidad ofrece un marco valioso [26].

Un ejemplo práctico de ello es la clasificación automática de documentos de texto, tratada como un problema de aprendizaje automático supervisado donde se utiliza el aprendizaje a partir de datos etiquetados para predecir la categoría de nuevos documentos. Según [27], en los modelos de aprendizaje automático buscan estimar una distribución de probabilidad P(y|x), y recurren a la teoría de la decisión para predecir el resultado más probable dada una entrada específica x.

2.4 Fases para clasificación de textos

Tras la revisión exhaustiva del estado del arte en la clasificación de textos utilizando PLN y aprendizaje automático supervisado, se han identificado y seleccionado los siguientes elementos para este estudio:

1. Algoritmos de clasificación:
   - Naïve Bayes (NB),
   - Bosques aleatorios (BA),
   - Regresión Logística (RL),
   - Máquinas de Soporte Vectorial (SVM),
   - k-Vecinos Más Cercanos (kNN),

   Estos algoritmos han sido elegidos por su eficacia demostrada en estudios previos y su equilibrio entre precisión, eficiencia y facilidad de implementación.
2. Técnicas de preprocesamiento de PLN:
   - Tokenización,
   - Conversión a minúsculas,
   - Lematización,
   - Eliminación de stopwords,
   - Eliminación de números, caracteres especiales y signos de puntuación.

   Estas técnicas han sido ampliamente validadas en la literatura como esenciales para mejorar la calidad de los datos y asegurar una interpretación más precisa de los textos.
3. Métodos de selección de características para la vectorización y sus valores:
   - Bolsa de palabras,
   - Frecuencia de Términos (TF),
   - TF-IDF,
   - Bigramas y trigramas.

   Estos métodos han demostrado ser eficaces para transformar el texto en formatos numéricos procesables por los algoritmos de aprendizaje automático, capturando la relevancia y relación entre las palabras.
4. Enfoques de entrenamiento:
   - División de datos 70-30 (70% entrenamiento, 30% validación),
   - Validación cruzada.

   Estos enfoques permiten una evaluación equilibrada y robusta de los modelos. El primero

es un método estándar en la comunidad de aprendizaje automático y el segundo, es una técnica que maximiza el uso de los datos disponibles para entrenamiento y validación garantizando una evaluación más fiable de los modelos.

5. Métricas de evaluación:
   - Matriz de confusión,
   - Precisión,
   - Recall,
   - Medida F.

Estas métricas son fundamentales para una evaluación exhaustiva del rendimiento de los modelos, proporcionando una comprensión integral de su precisión, sensibilidad y especificidad en la clasificación de textos.

2.5 Preprocesamiento en la Clasificación Automática de Textos

El preprocesamiento de textos es una etapa necesaria en el PNL, especialmente para la clasificación automática de textos. Esta fase implica la aplicación de diversas técnicas y procesos a los datos textuales antes de su análisis con algoritmos, preparando así los textos para una categorización precisa basada en su contenido [3]. El preprocesamiento transforma el texto original en una representación estructurada y normalizada, un paso esencial para construir modelos eficientes de clasificación textual.

Esta transformación, que mejora la calidad de los datos, reduce su dimensionalidad y complejidad, y facilita la detección de patrones, que resulta en un incremento significativo en la precisión y rendimiento de los clasificadores de textos [2, 3, 35]. Técnicas como la tokenización, lematización y eliminación de palabras irrelevantes juegan un papel importante en la normalización y preparación óptima de los datos textuales.

2.5.1 Extracción

La extracción de información es un proceso de selección y obtención de datos relevantes de conjuntos de datos específicos. Este proceso implica extraer información pertinente del conjunto de datos original y transformarla en documentos procesables, centrándose en el contenido más relevante para la posterior clasificación de textos [2,3]. Se emplean herramientas y técnicas automatizadas de extracción que son efectivas en datos textuales con cierta organización previa, con el objetivo de simplificar la estructura del texto original y convertirlo en una representación más adecuada para su análisis [28]

2.5.2 Tokenización

La tokenización es un proceso clave en el preprocesamiento de textos para el análisis y PLN. Consiste en segmentar un texto en unidades lingüísticas más pequeñas llamadas tokens, que pueden ser palabras individuales, signos de puntuación, números, etc. [3, 28]. Esta segmentación facilita tareas críticas como la detección de patrones, extracción de información, búsqueda y recuperación de textos, y es fundamental para realizar análisis textuales más complejos y avanzados [1].

2.5.3 Conversión a Minúsculas

La conversión de todo el texto a minúsculas es una técnica común de preprocesamiento. Permite la coincidencia de palabras independientemente de su capitalización original, facilitando así la búsqueda y recuperación de información. Esta normalización reduce la dimensionalidad del espacio vectorial y simplifica el procesamiento lingüístico, especialmente en idiomas con reglas complejas de [2, 3, 28]. Sin embargo, en algunos casos, es necesario aplicar heurísticas avanzadas para mantener distinciones útiles, como entre nombres propios y comunes.

2.5.4 Eliminación de Stopwords

La eliminación de stopwords, términos comunes, pero semánticamente vacíos, es fundamental para reducir el ruido en el procesamiento computacional de textos. Al excluir estas palabras de alta frecuencia, pero poco informativas, se disminuye el tamaño del diccionario en los sistemas de recuperación de información; mejorando así su eficiencia y rendimiento [2, 28, 29].

2.5.5 Eliminación de Signos de Puntuación, Caracteres Especiales y Números

Esta técnica de preprocesamiento implica la remoción de elementos no alfabéticos para simplificar y normalizar los textos. Al eliminar los signos de puntuación, símbolos y números, se centra la atención en las palabras relevantes y el contenido esencial del texto, lo que es crucial para el análisis semántico y las tareas específicas como la clasificación temática y el análisis de sentimientos [2, 3, 28].

2.5.6 Lematización

La lematización, un proceso de normalización léxica, reduce palabras a su forma canónica o lema, agrupando variantes flexionadas o derivadas bajo una forma estándar. Este proceso disminuye la dispersión léxica en la clasificación de textos y requiere un análisis morfológico y gramatical para identificar la categoría de cada palabra y extraer su base. Al reducir la redundancia en los datos, la lematización se vuelve un componente esencial en una variedad de tareas de PLN y aprendizaje automático [3, 28, 30].

En síntesis, el preprocesamiento de texto es una etapa inicial y fundamental en el Procesamiento del Lenguaje Natural (PLN), ya que desempeña un papel crucial en la preparación de datos de texto para su análisis y clasificación de manera eficaz. Las diversas técnicas de preprocesamiento contribuyen significativamente a la normalización y estandarización de los textos, abordando aspectos específicos, reduciendo la redundancia, simplificando la estructura y facilitando la identificación de elementos relevantes.

2.5.7 Métodos de Representación del Texto

Una vez abordadas las técnicas de preprocesamiento, se debe considerar la manera de transformar los textos preprocesados en formatos analizables por modelos de aprendizaje automático supervisados. Esto se logra mediante métodos de vectorización, entre los que destacan:

1. Bolsa de Palabras (BoW):

   La técnica de Bolsa de Palabras (BoW, por sus siglas en inglés) en el Procesamiento del Lenguaje Natural (PLN) implica la creación de un vector del tamaño del vocabulario, asignando valores desde uno hasta las posiciones correspondientes a las palabras presentes en un texto específico. A pesar de su simplicidad y eficiencia, esta técnica enfrenta varios desafíos significativos [31].

2. Frecuencia de Términos (TF):

   La Frecuencia de Términos (TF) es un concepto clave en el PLN, ya que mide cuántas veces un término específico, palabra o frase, aparece en un documento en comparación con el número total de términos en ese documento [32]; por ende, este enfoque asume que la relevancia de una palabra aumenta según su frecuencia de aparición, lo cual, matemáticamente, se expresa como la razón entre la cantidad de veces que el término de interés se encuentra en un documento y el número total de términos en ese mismo documento:

$$TF(t,d) = \frac{\#\ de\ veces\ que\ el\ término\ t\ aparece\ en\ el\ documento\ d}{número\ total\ de\ térmnos\ en\ el\ documento\ d}.$$

   En este caso “t” es el término de interés y “d” es el documento en cuestión.

   En concordancia, a Frecuencia de Términos (TF) se utiliza en diversas aplicaciones como: Motores de búsqueda, Sistemas de recomendación, Análisis de sentimientos y Clasificación de textos

   Su principal virtud radica en destacar la importancia de un término dentro de un documento específico. Sin embargo, esta métrica puede ser engañosa ya que presenta una limitación significativa cuando se aplica a colecciones de documentos, pues un término frecuente en un documento podría no ser distintivo si también es común en otros documentos de la colección.

3. TF-IDF (Frecuencia de Término - Frecuencia Inversa de Documento):

   La Frecuencia de Documento Inversa (IDF, por sus siglas en inglés) es una técnica utilizada en el PLN para evaluar qué tan único o raro es un término en una colección de documentos [19]. En términos matemáticos, esta frecuencia se calcula tomando el

logaritmo de la división del número total de documentos en la colección y se multiplica por el número de documentos en los que aparece el término de interés; además, es necesario sumar un uno al divisor, para evitar la división por cero:

$$IDF(t, D) = \log\left(\frac{documentos\ de\ la\ coleccion\ \ D}{documentos\ donde\ el\ termino\ t\ aparece\ +1}\right).$$

En esta fórmula, "t" representa el término cuya exclusividad se está evaluando, y "D" se refiere a la colección completa de documentos. Por lo tanto, la idea detrás de la IDF es:

- Disminuir la relevancia de términos frecuentes, pero que no contribuyen significativamente a diferenciar o distinguir un documento de otro.
- Otorgar mayor peso a términos menos comunes en toda la colección.
- Resaltar elementos verdaderamente distintivos y relevantes en cada texto.

En el ámbito del Procesamiento del Lenguaje Natural (PLN), la TF-IDF (Frecuencia de Términos - Frecuencia de Documento Inversa) es una técnica estadística muy utilizada para evaluar la relevancia de una palabra con respecto a una colección de documentos a la que pertenece [33, 35]. Esta técnica considera: La frecuencia con la que un término aparece en un documento (TF).

La fórmula de TF-IDF se expresa como:

$$TFIDF(t, d, D) = \mathrm{TF}(t, d) \times IDF(t, D).$$

Esta fórmula juega un papel importante en asignar relevancia a los términos dentro de un documento, otorgando valores más elevados de TF-IDF a aquellos términos que, siendo frecuentes en un documento específico, son raros en el corpus en su totalidad. Su aplicación es extensa en diversas áreas del PLN, incluyendo motores de búsqueda, generación de resúmenes automáticos, análisis de sentimientos y clasificación de textos [34].

4. N-gramas (bigramas y trigramas):
   Cada uno de estos métodos de vectorización presentan fortalezas y limitaciones específicas. Por ende, su elección depende del objetivo del análisis y las características del conjunto de datos. En el siguiente apartado, se profundizará en cómo estos métodos transforman el texto en vectores numéricos y su aplicación en el análisis y modelado en PLN.

La metodología n-gramas se distingue del modelo Bolsa de palabras por su habilidad para preservar la secuencia y el contexto en el que las palabras aparecen, lo cual permite identificar patrones lingüísticos más sofisticados y complejos [32]; por lo mismo, para la generación de n-gramas, se requiere emplear un procedimiento conocido como ventana deslizante que atraviesa el texto para agrupar los elementos en secuencias, cada una considerada un n-grama único.

2.6 Representación del texto

En el campo del PLN, la conversión de textos a formatos numéricos como vectores es esencial [31]; sin embargo, esta transformación es crucial no solo en el PLN, sino también en los algoritmos de Aprendizaje Automático y Aprendizaje Profundo. Como destacan [36], los documentos, son esencialmente secuencias de caracteres que deben adaptarse a un formato apropiado para su uso en algoritmos de clasificación automática.

Por tanto, el modelo vectorial se presenta como la técnica más utilizada para esta tarea, representando cada documento mediante un vector de términos con pesos asignados. Estos pesos reflejan la importancia o relevancia de cada término en el documento y pueden variar según el contexto y el campo de conocimiento del conjunto documentos de texto. En efecto, esta necesidad de transformación subraya la importancia de desarrollar técnicas eficientes para la extracción de características de los textos.

Sin embargo, el desafío más significativo en la representación de los textos consiste en asegurar representaciones similares para textos de temáticas afines, considerando:

a. La extensa variedad de vocabulario en cualquier lengua.
b. La tendencia de las representaciones textuales a ser dispersas.
c. La variación en tamaño de las representaciones según la longitud del texto.

Fig. 1. Metodología propuesta

Estos aspectos hacen énfasis en la importancia de adaptar las representaciones de los textos para mejorar su procesamiento y comprensión por las máquinas [31, 36].

3. Método

En este proyecto de investigación se lleva a cabo la clasificación automática del corpus Reuters-21578 mediante el manejo de técnicas de preprocesamiento y algoritmos clásicos de aprendizaje automático supervisado; para ello, se plantea el siguiente objetivo general.

Para PLN que integre técnicas de preprocesamiento y selección de características con algoritmos de aprendizaje automático supervisado, los cuales son ejecutados en python.

En este punto, se busca incrementar la precisión en la clasificación de textos del corpus Reuters-21578 en, al menos un 1%, mediante la aplicación de una metodología empírica rigurosa que permita evaluar y seleccionar una combinación efectiva de estas técnicas y algoritmos; lo cual, contribuirá significativamente al progreso de las áreas de PLN y aprendizaje automático. Cabe mencionar que con este módulo se busca perfeccionar y validar una metodología efectiva para la clasificación automática de documentos de texto que supere las métricas de rendimiento previamente establecidas.

3.1 Fases del proceso

Para alcanzar este objetivo, se siguen las fases del proceso de clasificación de textos con algoritmos de aprendizaje automático supervisado que consta de cinco etapas ver Fig.1.

Selección de datos

Para empezar, es importante tener en cuenta que cuando se está trabajando en tareas de PLN, concretamente en la clasificación automática de textos, es fundamental el empleo de un corpus estandarizado; este tipo de corpus se refiere a un conjunto de documentos que está al alcance de la comunidad interesada en la clasificación de textos. Mismo que será empleado por diversos investigadores a efectos de poder comparar los resultados con sus pares de una forma homogénea. Por lo mismo, la utilización de un corpus estandarizado, como Reuters-21578, permite a los investigadores comprobar los resultados de los diferentes modelos existentes y evaluar los propios, lo cual desemboca en proponer nuevas alternativas de clasificación.

El corpus Reuters-21578 Text Categorization Test Collection, es una colección de textos diseñada específicamente para ejecutar pruebas en el campo de la clasificación de texto. Aunque los derechos de autor pertenecen a Reuters Ltd., este conjunto de datos está disponible de forma gratuita para uso académico. Sin embargo, es importante destacar que cualquier publicación de resultados derivados de este corpus debe reconocer su uso y señalar la ubicación actual del conjunto de datos.

El corpus Reuters-21578 consta de artículos de periódico proporcionados por la agencia de noticias Reuters. Esta colección incluye un total de 21,578 archivos de texto, lo que da origen a su nombre. Los textos están disponibles en formato XML (eXtensible Markup Language), un lenguaje de marcado abierto que sigue un estándar derivado del SGML (Standard Generalized Markup Language) y que está optimizado para su uso en la web. Así en lugar de detallar las especificaciones del lenguaje XML, se proporciona un archivo como ejemplo para ilustrar su uso.

En los archivos del corpus, se utilizan las etiquetas SGML para estructurar la información de manera que se facilite su análisis y procesamiento. Estas etiquetas definen diferentes partes del contenido, como titulares, cuerpos de texto, fechas, entre otros elementos. De este modo, la estructura jerárquica de SGML asegura que los datos estén organizados de forma coherente y accesible.

3.2 Preprocesamiento

El proceso de selección de técnicas de preprocesamiento se basó en una revisión exhaustiva de la literatura y en experimentos preliminares realizados para evaluar su impacto en la calidad de los datos textuales. Estas técnicas fueron elegidas debido a su capacidad demostrada para normalizar y reducir el ruido en los textos como se mostrará en secciones siguiente.

Por tanto, el proceso de preprocesamiento en este proyecto se lleva a cabo en dos etapas clave. La primera etapa, implica la extracción y limpieza del texto que será utilizado en las experimentaciones. Este punto es vital para asegurar que los datos estén libres de impurezas o elementos irrelevantes que podrían afectar la calidad del análisis posterior.

La segunda etapa, por su parte, consiste en la aplicación de técnicas de preprocesamiento estandarizadas y comúnmente utilizadas en el campo. Estas técnicas incluyen, pero no se limitan a, la normalización de texto, la eliminación de palabras irrelevantes (stop words), la lematización y otras transformaciones que preparan el texto para un procesamiento más efectivo y eficiente por parte de los algoritmos de aprendizaje automático.

Este procedimiento se realiza en dos etapas:

1. La extracción y limpieza del texto (tratamiento del texto).
2. Aplicación de las técnicas de preprocesamiento estandarizadas. La cual contempla varios excenarios:
   - La data seleccionada se somete a la técnica de tokenización.
   - La data se procesa con las técnicas de tokenización y conversión a minúsculas.
   - Se aplica tokenización, conversión a minúsculas, eliminación de stopwords.
   - La data se trata con tokenización, conversión a minúsculas, eliminación de stopwords y eliminación tanto de signos de puntuación como caracteres especiales y lematización.
   - La data se trata con tokenización, conversión a minúsculas, eliminación de stopwords y eliminación tanto de signos de puntuación como caracteres especiales y lematización.

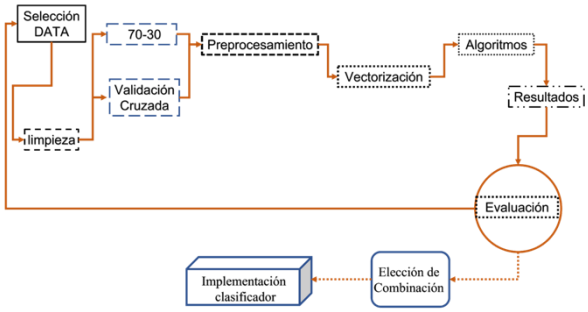

Fig. 2. Implementación metodológica

Ahora bien, cada una de estas fases integradas constituye la metodología de este módulo. En la Figura 2 se puede observar cómo la integración de cada una de las etapas completa el proceso para la clasificación automática de textos. Este enfoque permite visualizar la configuración con la mejor combinación de técnicas y algoritmos, lo que posibilita continuar con el objetivo principal de este trabajo. Por ende, la interacción entre las distintas fases desde la selección de la data hasta la evaluación de resultados proporciona un marco robusto de implementación para la clasificación automática. Así mismo, esta metodología no solo facilita la comprensión de cómo interactúan los diferentes componentes del sistema, sino que también contribuye al avance en el campo del Procesamiento del Lenguaje Natural y la clasificación automática de textos.

4. Resultados

Se presentan los resultados obtenidos para cada combinación experimental, centrándonos en la métrica de precisión que es el objeto de estudio de este trabajo. Estos resultados nos permitirán establecer conclusiones claras ya que se comparan favorablemente con los de estudios previos, al igual que se puede evidenciar en [6-7]. En particular, se observó que la integración de técnicas de preprocesamiento con algoritmos supervisados mejoró significativamente la precisión de la clasificación de textos. Por ejemplo,

mientras [7] reportaron una precisión del 95% utilizando SVM, nuestro enfoque logró una precisión de hasta el 98.6% en escenarios similares. Esta mejora puede atribuirse a la combinación específica de técnicas de preprocesamiento y métodos de selección de características realizada en este estudio. y definir los siguientes pasos en nuestra investigación.

Por tanto, es importante señalar que, para mantener la concisión y facilitar la lectura, este capítulo se enfoca exclusivamente en la métrica de evaluación de precisión. Sin embargo, se ha incluido el resultado de otras métricas relevantes (recall, puntaje F1) en los anexos. De esta manera, proporcionamos una visión completa y exhaustiva de nuestro estudio, permitiendo una comprensión más profunda del rendimiento de los modelos en diferentes aspectos.

Notación e Interpretación

Es importante resaltar que la presentación de los resultados obtenidos se encuentra relacionada con elementos que hacen facil la lectura y a su vez muestre la implementación de los algoritmos que se usaron para analizar los datos, como son el algorimo de K-vecinos más cercanos (kNN), Regresión Logística (LR), Naïve Bayes (NB), Random Forest (RF) y Máquinas de Soporte Vectorial (SVM):

- Ejes de las gráficas:*
- X: Métodos de selección de características,
- Y: Procesos de preprocesamiento,
- Z: Precisión obtenida,
- Colores de los algoritmos:*
- kNN: Azul,
- LR: Verde,
- NB: Rojo,
- RF: Cian,
- SVM: Magenta,
- Notación:*

V(algoritmo)#: V representa el método de selección de características (1-5), seguido del algoritmo. Ej: VKNN3 (TF-IDF para KNN)

p1-p5: Procesos de preprocesamiento

p1 ∩ VkNN3: Combinación específica de preprocesamiento P1, selección de características TF-IDF y kNN.

En cada gráfica, el punto con mayor precisión para cada algoritmo se resalta en negro. Los resultados de precisión para cada escenario experimental se muestran a continuación.

Para evaluar el rendimiento de los modelos, se aplicaron cinco procesos de preprocesamiento (p1-p5), cinco métodos de selección de características y representación de valores (v1-v5), y cinco algoritmos de aprendizaje automático supervisado, resultando en un total de 125 combinaciones posibles.

Asimismo, en cada uno de los 25 resultados por algoritmo, se ha etiquetado de color negro el puntaje donde se obtuvo el mayor porcentaje de acierto en la métrica de precisión.

Además, se utiliza la siguiente notación para representar cada combinación en el plano:

V(algoritmo)#: Donde V indica el método de selección de características, seguido del algoritmo y un número que representa la característica (1: Bolsa de Palabras, 2: Frecuencia de Término, 3: TF-IDF, 4: Bigramas, 5: Trigramas). Por ejemplo, VKNN3 representa el método TF-IDF para KNN.

La intersección de un proceso de preprocesamiento (p1-p5) con una combinación V(algoritmo)# indica la combinación específica de preprocesamiento, selección de características y algoritmo utilizado. Por ejemplo, p1 ∩ VkNN3 representa el escenario de preprocesamiento P1 con TF-IDF y kNN. A continuación, se presentan los resultados de precisión obtenidos para cada escenario experimental realizado, en este aparatdo se presenta el mejor resultado obtenido. En este proceso se presentan 4 escenarios de ejecuciones, el primero es una ejecución de datos con un 70% de los datos para entrenamiento y el 30% de los datos para prueba, el segundo escenario es una validación cruzada, al igual que se realiza División 70/30 con datos aumentados y la última validación es Validación Cruzada con datos aumentados, en los cuales se puede observar, que las configuraciones establecidas en este trabajo, cuenta con una precisión superior a las implementadas por los investigadores que se relacionan este trabajo. Es por ello que se presenta el escenario 3 que es División 70/30 con datos aumentados, la cual presenta la mejor precisión entre todas las configuraciones.

El Escenario 1 se basa en un conjunto de datos que abarca 10 categorías temáticas (acq, coffee,

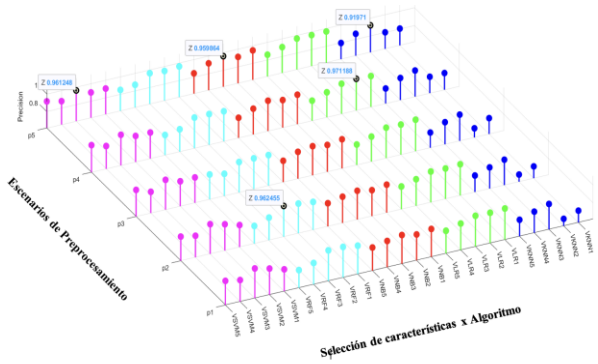

Fig. 3. Resultados de Precisión para las 125 Combinaciones del Escenario 1

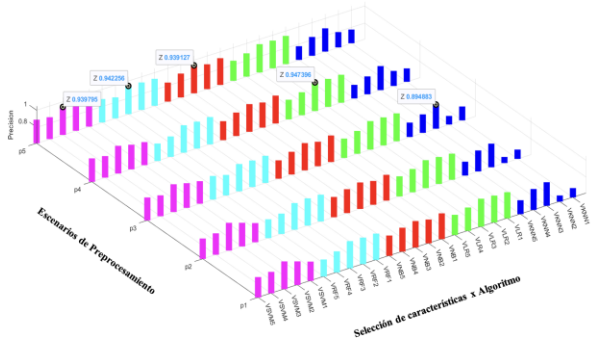

Fig. 4. Resultados de Precisión para las 125 Combinaciones del Escenario 3

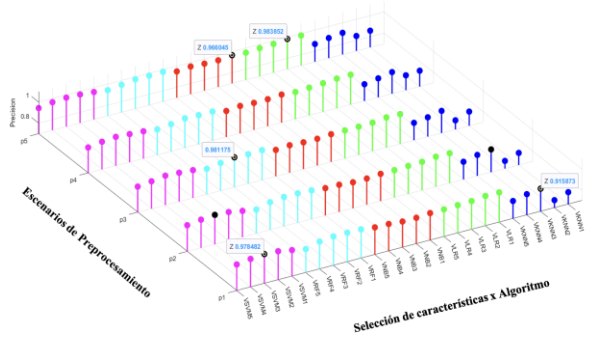

Fig. 5. Resultados de Precisión para las 125 Combinaciones del Escenario 3

crude, earn, gold, interest, money-fx, ship, sugar, trade), cada una con 100 documentos. Se realizó una división 70/30 de los datos, utilizando el 70% para entrenamiento y el 30% restante para validación. A continuación, se presentan los resultados de precisión obtenidos para las 125 combinaciones posibles en este escenario.

El Escenario 2 se basa en el mismo conjunto de datos del Escenario 1, abarcando las 10 categorías temáticas (acq, coffee, crude, earn, gold, interest, money-fx, ship, sugar, trade), cada una con 100 documentos. En este caso, se empleó el método de validación cruzada con 5 particiones en lugar de una división fija 70/30. La validación cruzada se utilizó para asegurar que los resultados fueran más robustos y menos dependientes de una única partición de los datos, proporcionando una evaluación más generalizada del rendimiento de los modelos.

El Escenario 3 amplía el conjunto de datos del Escenario 1, manteniendo las mismas 10 categorías temáticas (acq, coffee, crude, earn, gold, interest, money-fx, ship, sugar, trade), pero aumentando el número de documentos por categoría a 500.

Para lograr esto, se siguieron los siguientes criterios:

- Categorías con más de 500 documentos: Se seleccionaron aleatoriamente 500 documentos sin repetición.
- Categorías con menos de 500 documentos: Se incluyeron todos los documentos disponibles y se completaron los 500 restantes mediante la repetición aleatoria de los documentos originales. Por ejemplo, en la categoría “sugar” con 500 documentos, se seleccionaron aleatoriamente los 500 documentos cinco veces para alcanzar los 500 requeridos.

En este escenario, también se realizó una división 70/30 de los datos, utilizando el 70% para entrenamiento y el 30% restante para validación. A continuación, se presentan los resultados de precisión obtenidos para las 125 combinaciones posibles en este escenario, Precisión según Escenarios de Preprocesamiento y Selección de Características por Algoritmo Escenario 3.

El Escenario 4 se utiliza el mismo el conjunto de datos del Escenario 3, abarcando las mismas 10 categorías temáticas (acq, coffee, crude, earn, gold, interest, money-fx, ship, sugar, trade), cada conformada por 500 documentos; no obstante la diferencia clave radica en el método de evaluación, puesto que, en lugar de una división

Tabla 1. Resultados del Escenario 4-Validación Cruzada con datos aumentados

Algoritmo	Precisión	Mejor combinación
K-Vecinos más Cercanos	90,29%	Preprocesamiento: [P2], Vectorización: [TV3]
regresión logistica	96,63%	Preprocesamiento: [P5], Vectorización: [TV1]
Naïve Nayes	94,35%	Preprocesamiento: [P4], Vectorización: [TV1]
Random Forest	95,97%	Preprocesamiento: [P4], Vectorización: [TV3]
Máquinas de Soporte Vectorial	95,40%	Preprocesamiento: [P5], Vectorización: [TV3]

fija 70/30, en este escenario se emplea validación cruzada con 5 particiones para obtener una evaluación más robusta y generalizada en torno al rendimiento de los modelos reduciendo la dependencia de una división de los datos única y proporcionando un aproximado más preciso respecto a la capacidad de generalización de los modelos.

5. Discusión de Resultados

Este estudio se centró en la optimización de la clasificación automática de textos para el corpus Reuters-21578, evaluando diversos algoritmos, técnicas de preprocesamiento y métodos de vectorización. A continuación, se discuten los hallazgos principales y sus implicaciones. Es de resaltar que las ejecuciones realizadas y los resultados obtenidos estan articuladas para realizar las pruebas y trazabilidad que presenta el algoritmo propuesto en el corpus. Se puede evidenciar que el ajuste presentado genera una precisión superior en todos los procesos, mucho más elevado a lo presentado en los articulos relacionados en esta bibliografia.

En este estudio se evaluaron dos métodos: división de datos 70/30 y validación cruzada en dos conjuntos de datos diferentes Data_100 y Data_500. El primero con 100 documentos por categoría mientras que el segundo con 500 documentos por categoría. En primer lugar, la división 70/30 mostró generalmente valores más altos de precisión. Por otro lado, la validación cruzada proporcionó estimaciones más conservadoras del rendimiento de los modelos. Por ejemplo, mientras que para el método 1 la precisión fue del 97,12%, el otro mostro una precisión del 94,74%.

En cuanto al impacto del aumento de datos, los escenarios 3 y 4, que utilizaron el conjunto de datos ampliado Data_500 mostraron una mejora significativa en la precisión en comparación con los escenarios 1 y 2 que utilizaron Data_100. Este aumento en la precisión demuestra que la ampliación del conjunto de datos es un factor que puede influir considerablemente para mejorar el rendimiento de los modelos de clasificación de textos. Por ejemplo, la precisión del modelo Random Forest aumentó del 96.25% en el Escenario 1 al 98.12% en el Escenario 3, lo que resalta la importancia de un mayor volumen de datos para un mejor desempeño.

Los mejores resultados presentados en los procedimientos presentados anteriormente son los realizados con las Máquinas de Soporte Vectorial (SVM) también demostraron un desempeño sólido y consistente a través de todos los escenarios. Logrando un porcentaje de precisión del 97.85% con la Data_500. Para este algoritmo el preprocesamiento fue el completo denotado como P5 y la vectorización TF-IDF, lo que le permitió manejar de manera efectiva los datos textuales.

En cuanto a las técnicas de preprocesamiento, se encontró que el escenario P5, el cual incluye las técnicas de tokenización, conversión a minúsculas, eliminación de stopwords, eliminación de puntuación/caracteres especiales y lematización, fue continuamente la opción más efectiva. Esto sugiere que una limpieza exhaustiva y una normalización de los datos textuales contribuyen significativamente al rendimiento de los modelos de clasificación. Sin embargo, los escenarios P4 y P2 también mostraron buenos resultados, lo que indica que la lematización no siempre es necesaria para lograr un buen rendimiento.

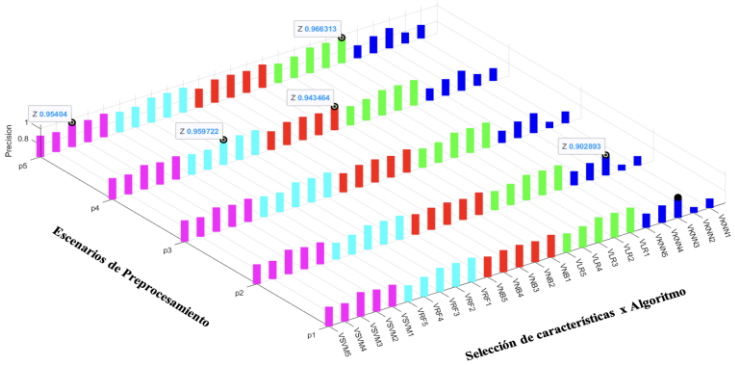

Fig. 6 Resultados de Precisión para las 125 Combinaciones del Escenario 3

Tabla 2. Resultados del Escenario 1 -División 70/30

Algoritmo	Precisión	Mejor Combinación
K-Vecinos más Cercanos	91,97%	Preprocesamiento: [P5], Vectorización: [TV3]
Regresión logística	97,12%	Preprocesamiento: [P4], Vectorización: [TV2]
Naïve Bayes	95,99%	Preprocesamiento: [P5], Vectorización: [TV3]
Random Forest	96,25%	Preprocesamiento: [P2], Vectorización: [TV3]
Máquinas de Soporte Vectorial	96,12%	Preprocesamiento: [P5], Vectorización: [TV3]

Tabla 3. Resultados del Escenario 2-Validación cruzada

Algoritmo	Precisión	Mejor Combinación
K-Vecinos más Cercanos	89,49%	Preprocesamiento: [P3], Vectorización: [TV3]
Regresión logística	94,74%	Preprocesamiento: [P4], Vectorización: [TV3]
Naïve Bayes	93,91%	Preprocesamiento: [P5], Vectorización: [TV3]
Random Forest	94,23%	Preprocesamiento: [P2], Vectorización: [TV3]
Máquinas de Soporte Vectorial	93,98%	Preprocesamiento: [P5], Vectorización: [TV3]

Fuente: Elaboración Propia

Tabla 4. Resultados del Escenario 3 -División 70/30 con datos aumentados

Algoritmo	Precisión	Mejor Combinación
K-Vecinos más Cercanos	91,59%	Preprocesamiento: [P2], Vectorización: [TV3]
Regresión logistica	98,39%	Preprocesamiento: [P5], Vectorización: [TV2]
Naïve Nayes	96,60%	Preprocesamiento: [P5], Vectorización: [TV1]
Random Forest	98,12%	Preprocesamiento: [P3], Vectorización: [TV3]
Máquinas de Soporte Vectorial	97,85%	Preprocesamiento: [P2], Vectorización: [TV3]

Fuente: Elaboración Propia

Esta observación destaca la importancia de ajustar las técnicas de preprocesamiento según el contexto y el tipo de colección de datos.

En términos de vectorización, el método TF-IDF se destacó como el más efectivo en la mayoría de los casos. La capacidad de TF-IDF para capturar la relevancia de las palabras en los documentos resultó fundamental para mejorar la precisión de los modelos de clasificación. No obstante, otros métodos de vectorización como la frecuencia de términos y la bolsa de palabras también demostraron ser útiles en ciertos contextos, ofreciendo buenos resultados. Esta variedad en la efectividad de las técnicas de vectorización recalca la necesidad de experimentar con diferentes enfoques para encontrar la combinación adecuada para cada aplicación específica.

Por tanto, a mejor combinación de escenario de procesamiento, método de vectorización y algoritmo de aprendizaje supervisado obtenida para este estudio es la siguiente:

- Algoritmo: Regresión Logística
- Precisión: 98.39%
- Preprocesamiento: P5
- Vectorización: Frecuencia de términos
- Escenario: División de Datos 70/30 con Data_500

El análisis detallado de esta combinación revela que el escenario de preprocesamiento P5 resultó ser el más efectivo para este corpus específico. Sorprendentemente, la vectorización basada en la frecuencia de términos superó a TF-IDF, lo que sugiere que la frecuencia bruta es un indicador más efectivo para este conjunto de datos. La Regresión Logística demostró ser particularmente adecuada para la naturaleza de los textos en Reuters-21578, logrando una precisión del 98.39%.

Así mismo, La influencia del tamaño del conjunto de datos también fue evidente en los resultados obtenidos. La combinación seleccionada se logró en el escenario con datos aumentados Data 500, lo que marca la importancia de contar con un conjunto de datos más amplio para mejorar la precisión de la clasificación.

6. Conclusiones

Este estudio ha identificado una combinación de técnicas y un algoritmo que ofrecen un rendimiento mejorado en la clasificación de textos del corpus Reuters-21578. Además, la precisión obtenida supera ampliamente el objetivo inicial, proporcionando una base sólida para futuros trabajos de investigación y aplicaciones prácticas en la clasificación de textos similares.

Un hallazgo importante para destacar es la mejora notable en los resultados obtenidos a partir del uso de la data aumentada. En este sentido, este descubrimiento subraya la importancia de la disponibilidad y volumen de datos a la hora de entrenar los modelos de aprendizaje supervisado, lo que implica obtener resultados con mayor precisión.

Adicionalmente, se observa una interesante relación inversa entre el tamaño del conjunto de datos y la necesidad de técnicas de preprocesamiento. Esta relación mostró que a medida que aumentaba el volumen de datos, se requería una menor cantidad de técnicas de preprocesamiento para lograr mayores resultados en la métrica de precisión para algunos algoritmos. Por ende, esta observación sugiere que un conjunto de datos más grande puede compensar, hasta cierto punto, la necesidad de un preprocesamiento excesivo. Sin embargo, es importante señalar que esta relación entre el tamaño de los datos y la reducción en el preprocesamiento es una condición suficiente pero no necesaria para obtener mejores resultados. Es otras palabras, aunque un conjunto de datos más grande puede permitir un preprocesamiento menos intensivo, esto no garantiza automáticamente resultados superiores en todos los casos.

La presente investigación se propuso desarrollar un programa en Python para Procesamiento del Lenguaje Natural (PLN) que integrara técnicas de preprocesamiento y selección de características con algoritmos de aprendizaje automático supervisado, con el objetivo de mejorar la precisión en la clasificación de textos del corpus Reuters-21578. A lo largo de este estudio, se ha logrado no solo cumplir sino superar significativamente este objetivo, obteniendo resultados que contribuyen de manera

sustancial al campo del PLN y la clasificación automática de textos.

References

1. **Aggarwal, C.C., Zhai, C. X. (2012).** A survey of text classification algorithms. Mining Text Data, pp. 163–222, Springer, DOI: 10.1007/978-1-4614-3223-4_6
2. **Manning, C.D., Raghavan, P., Schütze, H. (2009)**. An Introduction to Information Retrieval. Cambridge Cambridge UP, Cambridge University Press, http://www.informationretrieval.org/
3. **Jurafsky, D., Martin, J.H. (2023)**. Speech and Language Processing an Introduction to Natural Language Processing. Computational Linguistics, and Speech Recognition, D. Jurafsky & D. Jurafsky, Vol. 1, Pearson.
4. **Kowsari, K., Meimandi, K.J., Heidarysafa, M., Mendu, S., Barnes, L., Brown, D. (2019)**. Text classification algorithms: A survey. Information (Switzerland), Vol. 10, No. 4, MDPI AG. DOI: 10.3390/info10040150.
5. **Chen, R.C., Dewi, C., Huang, S.W., Caraka, R.E. (2020)**. Selecting critical features for data classification based on machine learning methods. Journal of Big Data, Vol. 7, No. 1. DOI: 10.1186/s40537-020-00327-4.
6. **Pais, S., Cordeiro, J., Jamil, M.L. (2022)**. NLP-based platform as a service: A brief review. Journal of Big Data, Vol. 9, No. 1. DOI: 10.1186/s40537-022-00603-5.
7. **Dhar, A., Mukherjee, H., Dash, N.S., Roy, K. (2021).** Text categorization: past and present. Artificial Intelligence Review, Vol. 54, No. 4, pp. 3007–3054, DOI: 10.1007/s10462-020-09919-1.
8. **Kadhim, A.I. (2019)**. Survey on supervised machine learning techniques for automatic text classification. Artificial Intelligence Review, Vol. 52, No. 1, pp. 273–292. DOI: 10.1007/s10462-018-09677-1.
9. **Alloghani, M., Al-Jumeily, D., Mustafina, J., Hussain, A., Aljaaf, A.J. (2020)**. A Systematic Review on Supervised and Unsupervised Machine Learning Algorithms for Data Science, Vol. 3–21, Springer. DOI: 10.1007/978-3-030-22475-2_1.
10. **Shah, K., Patel, H., Sanghvi, D., Shah, M. (2020).** A Comparative Analysis of Logistic Regression, Random Forest and KNN Models for the Text Classification. Augmented Human Research, Vol. 5, No. 1. DOI: 10.1007/s41133-020-00032-0.
11. **Chen, Z., Zhou, L.J., Li, X. Da, Zhang, J.N., Huo, W. J. (2020)**. The Lao text classification method based on KNN. Procedia Computer Science, Vol. 166, pp. 523–528. DOI: 10.1016/j.procs.2020. 02.053.
12. **Bhavani, A., Santhosh Kumar, B. (2021)**. A Review of State Art of Text Classification Algorithms. Proceedings - 5th International Conference on Computing Methodologies and Communication, ICCMC, pp. 1484–1490. DOI: 10.1109/ICCMC51019.2021.9418262.
13. **Shah, K., Patel, H., Sanghvi, D., Shah, M. (2020).** A Comparative Analysis of Logistic Regression, Random Forest and KNN Models for the Text Classification. Augmented Human Research, Vol. 5, No. 1, DOI: 10.1007/s41133-020-00032-0.
14. **Singh, J., Gupta, V. (2016).** Text stemming: Approaches, applications, and challenges. ACM Computing Surveys, Vol. 49, No. 3. DOI: 10.1145/2975608.
15. **Zhao, W., Chen, J.J., Perkins, R., Liu, Z., Ge, W., Ding, Y., Zou, W. (2015)**. A heuristic approach to determine an appropriate number of topics in topic modeling. BMC Bioinformatics, Vol. 16, No. 13. DOI: 10.1186/1471-2105-16-S13-S8.
16. **Li, Y. (2021)**. Automatic Classification of Chinese Long Texts Based on Deep Transfer Learning Algorithm. 2021 2nd International Conference on Artificial Intelligence and Computer Engineering (ICAICE), pp. 17–20. DOI: 10.1109/ICAICE54393. 2021.00011.
17. **Duong, H.-T., Nguyen-Thi, T.-A. (2021)**. A review: Preprocessing techniques and data augmentation for sentiment analysis. Computational Social Networks, Vol. 8, No. 1. DOI: 10.1186/s40649-020-00080-x.
18. **Zhang, C., Wang, H., Liu, Y., Wu, D., Liao, Y., Wang, B. (2008)**. Automatic Keyword

Extraction from Documents Using Conditional Random Fields. Journal of Computational Information, Vol. 4. http://www.JofCI.org1553-9105/

19. **Sun, S., Luo, C., Chen, J. (2017).** A review of natural language processing techniques for opinion mining systems. Information Fusion, Vol. 36, pp. 10–25. DOI: 10.1016/J.INFFUS.2016.10.004.
20. **Huang, Y., Chen, J., Zheng, S., Xue, Y., Hu, X. (2021)**. Hierarchical multi-attention networks for document classification. International Journal of Machine Learning and Cybernetics, Vol. 12, No. 6, pp. 1639–1647. DOI: 10.1007/s13042-020-01260-x
21. **Sebastiani, F. (2002)**. Machine Learning in Automated Text Categorization. ACM Computing Surveys, Vol. 34, pp. 1–47. www.ira.uka.de/bibliography/Ai/automated. text.
22. **Cárdenas, J., Olivares, G., Alfaro, R. (2014)**. Clasificación automática de textos usando redes de palabras. Revista Signos, Vol. 47, No. 86, pp. 346–364. DOI: 10.4067/S0718-09342014000300001.
23. **Guardiola González, C. (2020).** Clasificador de textos mediante técnicas de aprendizaje automático. https://riunet.upv.es:443/handle/10251/ 133840.
24. **Patel, A., Pathak, S., Khan, M.I. (2021)**. Automated Text Categorization. 2021 3rd International Conference on Signal Processing and Communication (ICPSC), pp. 16–20. DOI: 10.1109/ICSPC51351.2021.9451670.
25. **Mitchell, T.M. (1997)**. Machine Learning. Vol. 1, McGraw-Hill Science/Engineering/Math.
26. **Quirós Díaz, L., Vidal, E. (2021)**. Layout Analysis for Handwritten Documents a Probabilistic Machine Learning Approach. Tesis de Doctorado, Universitat Politécnica de Valéncia.
27. **Bishop, C.M. (2006)**. Pattern Recognition and Machine Learning. Vol. 1, Springer.
28. **Weiss, S.M., Indurkhya, N., Zhang, T., Damerau, F. J. (2005)**. Text Mining: Predictive Methods for Analyzing Unstructured Information. In **S. M. Weiss, N. Indurkhya, T. Zhang, Fred J. Damerau**, editors, Vol. 1. Springer Science+Business Media.
29. **Mohan, V., Ilamathi, J. (2020).** Preprocessing Techniques for Text Mining-An Overview. International Journal of Computer Science & Communication Networks, Vol. 5, No. 1, pp. 7–16. https://www.researchgate.net/publication/339529230.
30. **González Escudero, V., Parapar López, J. (2022).** Plataforma para Procesado de Lenguaje Natural como Servicio. Tesis de pregrado, Universidad de Coruña.
31. **Giménez Fayos, M.T. (2021).** Natural Language Processing using Deep Learning in Social Media. Universitat Politècnica de València. DOI: 10.4995/Thesis/10251/172164.
32. **Ruiz Rico, F. (2013)**. Selección y ponderación de características para la clasificación de textos y su aplicación en el diagnóstico médico. Tesis Doctoral, Universidad de Alicante.
33. **Dalaorao, G.A., Sison, A.M., Aguinaldo, E., Medina, R. P. (2019)**. Integrating Collocation as TF-IDF Enhancement to Improve Classification Accuracy. IEEE 13th International Conference on Telecommunication Systems, Services, and Applications (TSSA), Vol. 1, No. 1, pp. 282–285. DOI: 10.1109/TSSA48701.2019.8985458.
34. **Rahmah, A., Santoso, H.B., Hasibuan, Z.A. (2019)**. Exploring Technology-Enhanced Learning Key Terms using TF-IDF Weighting. IEEE, pp. 1–4. DOI: 10.1109/ICIC47613.2019.8985776.
35. **Sidorov, G. (2019).** Syntactic n-grams in computational linguistics. Springer, 98 p.
36. **Lojo Vicente, J.D., Barreiro García, Á., Losada Carril, D. (2012).** Tesis Doctoral Clasificación Automática de Documentación Clínica. Tesis Doctoral, Universidad de Coruña.

Article received on 30/10/2022; accepted on 17/01/2025.
**Corresponding author is Rafael Guzmán Cabrera.*